Grant Agreement No.: 261527

Mantychore

IP Networks as a Service

Instrument**: *Combination of Collaborative Project and Coordination and Support Action***

Thematic Priority: **[INFRA-2010.1.2.3] Virtual Research Communities**

D7.1Integration between Mantychore and GSN report

Due date of the Deliverable: 31st, January 2012

Actual submission date: 13/09/2012

Start date of project: October 1st 2010 Duration: 30 months

Project Manager: Sergi Figuerola (i2CAT)

Version: 1.0

Deliverable leader: NORDUnet

Author List: Alin Pastrama (NORDUnet), Andrew Mackarel (HEAnet),
Pau Minoves (i2CAT), Isart Canyameras (i2CAT)

Project co-funded by the European Commission in the 7th Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract

This report describes the work and outcomes of the integration between the Mantychore and GreenStar Network (GSN) projects. The document describes the network integration activity with all its steps and phases, as well as all aspects of the service integration work and its results. It also contains information pertaining to the liaisons and collaborations with external parties.

Table of Contents

1 Executive Summary 1
2 Objectives and Scope 2
3 Introduction 4
4 Background and Related Work 5
4.1 Mantychore and OpenNaaS 5
4.2 GreenStar Network 5
5 Network Integration 7
5.1 GSN Nodes 7
5.1.1 HEAnet Nodes 9
5.1.2 i2CAT Node 12
5.1.3 IBBT Node 12
5.1.4 NORDUnet Node 12
5.2 Mantychore/GSN Network 13
5.2.1 Phase 1 13
5.2.2 Phase 2 14
5.2.3 Phase 3 15
5.2.4 Current Status 16
6 Service Integration 18
6.1 GSN Use Case 18
6.1.1 Long-term Connectivity 18
6.1.2 Temporary Connectivity 19
6.2 API Design 21
6.3 Integration of Power Monitoring Metrics 22
6.3.1 Overview 22
6.3.2 OpenNaaS Power Monitoring and Management API 23
6.3.3 Portability details 24
6.3.4 Future extensions 25
7 Liaisons 26
8 Conclusion 27
9 References 29
10 Acronyms 30
11 Appendices 31

11.1 Solar Power Generation at EPA.....31
11.2 GSN Host Installation Process.....31

Figure Summary

Figure 1 - GSN nodes in Europe 8
Figure 2 - Solar panel array at the HEAnet EPA site 9
Figure 3 - HEAnet GSN node at EPA 10
Figure 4 - HEAnet GSN node at DKIT 11
Figure 5 – Wind Turbine at the HEAnet DKIT site 11
Figure 6 - NORDUnet GSN node 13
Figure 7 - Mantychore/GSN network - Phase 1 14
Figure 8 - Mantychore/GSN network - Phase 2 15
Figure 9 - Mantychore/GSN network - Phase 3 16
Figure 10 - Mantychore/GSN network - Current status 17
Figure 11 - GSN use case for long term connectivity 19
Figure 12 - GSN use case for temporary connectivity 20
Figure 13 - OpenNaaS - GSN messaging sequence 22

1 Executive Summary

This deliverable describes the integration between the Mantychore and GSN projects. The report is split into two main sections: network integration and service integration.

The section pertaining to the integration of the network – or physical – layer starts by focusing on the deployment of GSN nodes, in Europe, by the WP7 partners. These nodes consist of processing, storage, networking and measurement equipment powered by various renewable energy sources (in most of the cases). This section also discusses and explains the network connectivity, as the connectivity between the GSN nodes in Europe to those in Canada . It follows a set of predefined steps taken with the purpose of obtaining an optimal and resilient network infrastructure. The chapter itself is dedicated to network integration ends with a snapshot of the status of the integrated Mantychore/GSN network at the time of writing this deliverable.

The second major section of this report discusses the service's – or software – cooperation between the two projects. This chapter begins with identifying the use cases where interoperability at the software level would be required. Once these use cases are defined, a high-level design for an API is described with the purpose of acting as a specification for a future OpenNaaS development task. The section concludes with a discussion on the integration of specific Green GSN features within the OpenNaaS information model – namely the power monitoring facilities for network devices.

Finally the report ends with a review of the liaisons and collaborations involved in the work package so far, and a sum-up of what was achieved and the work ahead.

2 Objectives and Scope

As per the Mantychore project's Description-of-Work document [1]:

MANTYCHORE will start collaborations with GSN (GreenStar Network) project for the integration of its NREN infrastructures in its test-bed where GSN network is formed by a set of green nodes where each node is powered by sun, wind, etc. GSN Carbon Measurement Protocol is used for GSN to measure the quantity of energy and carbon emission permitted in these nodes allowing for better management. It will be utilized to generate carbon credits from the reduction of carbon resulting from relocation different network resources among networks and test beds. Relocating these resources could possibly save tons of carbon per year.

The NREN infrastructure will be connected to the GSN network and it will add new resources, as Mantychore resources. Currently Mantychore features allow new possibilities to solve different issues in configuration, testing, monitoring, etc.

Benefits of the proposed User case studies are:

- *The location of the consortium and its distance from other sources such as the GSN test bed from Canada and Mantychore infrastructure at Europe adds new interesting user cases to this project. These new user cases analyze how it is possible to move services to computing or networking resources and why its situation and location can influence that choice.*
- *To collaborate on cloud management algorithms making optimal use of the intermittently available renewable energy sources.*
- *The viability and usefulness that will be measured by the degree to which the renewable energy sources can be utilized. This usefulness will be checked in the different use cases and will specify the best options to reduce emissions and energy.*
- *To monitor different actions and movement of services among sources. These workflow and goals will be transparent for end users who will not see any difference and will not have any interruption in their tasks. For an end user, his service will work continuously without any problem.*

The main goal of this task is to setup a GSN node that will allow access from varied NREN infrastructures to the GSN network. To achieve this subtasks are required to carry out:

- *Analysis of different possibilities to connect the two infrastructures.*
- *Integration of Mantychore physical resources in GSN. It includes all the process of installation of a connection Mantychore-GSN.*
- *To specify requirement and planning of the software integration. It is necessary understand and specify how Mantychore and GSN services will be able to collaborate.*
- *Mantychore services provide facilities to configure routers and interfaces and to give flexibility thanks to its IaaS architecture and management of logical devices. GSN profits from all these features to improve its infrastructure.*

At the time of this deliverable, and due to the evolution of the GSN project, there has been a new amendment for the Mantychore project. In this amendment, it has been agreed that besides integrating the two software stacks of both project, to extend the Mantychore's information model with the Green parameters considered within the GSN project, and also to develop an API that will allow GSN partners, and those on the new consortium (ECOLOTIC) as agreed on the new MoU signed with them.

3 Introduction

The Infrastructure as a Service (IaaS) model has gained popularity in recent years due to the flexibility, scalability and reliability that it offers to companies and institutions that do not have the means or the interest to manage their own network infrastructures. IaaS concepts are allowing nowadays the offering of features such as distributed computing, network virtualisation, media distribution and energy efficiency. At the forefront of IaaS development are the research communities around the world, with national and regional research and education networks (RENs) as their enablers.

Research communities in Europe and around the world often work in a distributed fashion with geographically distant research centers, data producers and consumers. As a consequence, they require advanced network infrastructures that meet their demands for performance, efficiency and flexibility. In fact, these kind of network services are not yet widely deployed from commercial operators, and if so they are prohibitively expensive due to the amount manual work and dedicated resources that they imply; this fact warrants the need for intelligent, open and flexible network provisioning solutions.

Moreover, energy efficiency and carbon reduction are factors that are important for both the providers and the end users of network services. The providers are interested in increasing their energy efficiency and thus decreasing infrastructure costs, while the end users want their services to be carbon neutral in order to comply with industry standards such as the ISO 14064 series.

Two NREN-driven IaaS projects are Mantychore and GreenStar Network (GSN) & ECOLOTIC, which deal with network virtualisation (providing IP networks as a service) and increasing the use of renewable energy in the ICT sector, respectively. The goal of the present work, as part of the Mantychore project, is to study and plan for interoperability of these two projects in order to expand their service and feature portfolios.

4 Background and Related Work

4.1 Mantychore and OpenNaaS

The Mantychore project aims to allow the national RENs (NRENs) to provide a complete, flexible IP network service that allows research communities to have control of their network infrastructures (i.e. layer 3 IP addressing, internal routing, peering, firewalls) [1,2]). The concept of Network as a Service (NaaS) targets the virtualisation of network resources and it is a subset of Infrastructure as a service IaaS. In the cloud-computing paradigm, networks are provisioned automatically, just like computing resources, but also require handling of standardisation and vendor detail abstraction. Service consumers now have complex equipment setups, which require a level of direct management over such networking resources. OpenNaaS is the software tool developed in the Mantychore project and it aims to become a modern and flexible NaaS toolset that empowers IaaS service providers with increased and sophisticated management flexibility on the network.

4.2 GreenStar Network

The GSN project was funded by CANARIE, the Canadian NREN, with the goal to create technology and standards to reduce the carbon footprint of ICT. The project is based around three main components: a networking and computational infrastructure, a middleware developed to provide cloud services and applications, and a carbon protocol based on the ISO 14064 family of standards [3].

The GSN implementation combined the three aforementioned aspects to migrate virtual machines (VMs) between geographically distributed datacenters, or GSN nodes, based on the availability of renewable energy [4,5]. Currently the GSN core network consists of a hub node and a number of spoke nodes placed at different locations in Canada, within the CANARIE network; the extended network includes the core network, plus spoke nodes located in the Unites States and Europe, operated in collaboration with the local RENs.

The current GSN project has finished, however, the new phase of the project, called ECOLO TIC, is leveraging the outcomes of GSN in a wider scope.

5 Network Integration

The network integration between the Mantychore and GSN projects consisted in the deployment of a number of GSN nodes by the partners involved in the work package. These nodes, built following the GSN specifications, were first connected to the Canadian GSN network services and later reorganised in a European GSN small network.

5.1 GSN Nodes

The GSN nodes deployed by the WP7 partners consist of the following elements:

- One or more sources of renewable energy
- A server
- A switch or router
- One or more PDUs

The renewable energy sources used by the European GSN nodes are wind power, solar power, and hydro- and geothermal energy. The servers closely followed the specifications received from the GSN infrastructure team, based on the Dell PowerEdge R710 series. The networking devices used, whose purpose is to provide connectivity to the GSN cloud, were various Cisco Systems, Juniper Networks and SMC models. Finally, the PDUs also varied, with models from Raritan, Avocent and Eaton being employed.

Figure 1 below depicts the geographic distribution of the GSN nodes deployed by WP7 partners.

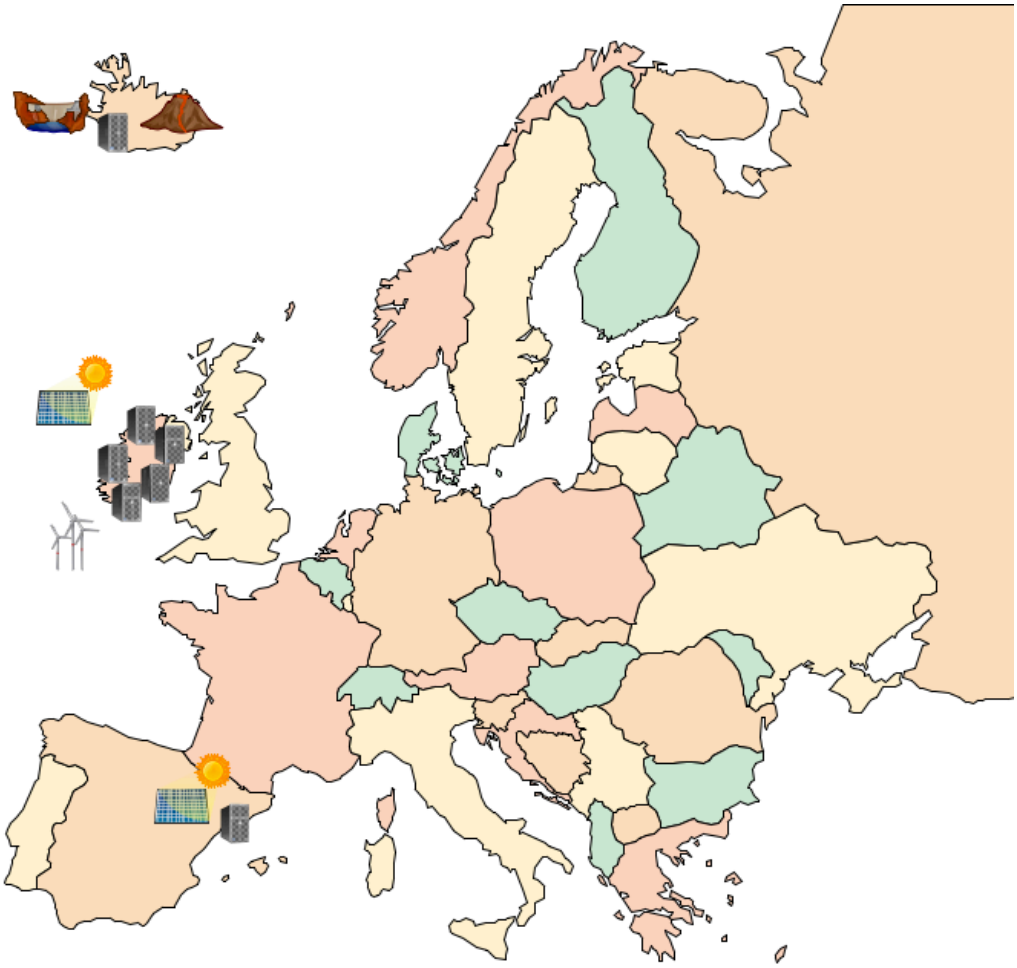

Figure 1 - GSN nodes in Europe

Software-wise, the GSN node servers are configured as follows:

1. Install operating system Ubuntu 10.04 64-bit
2. Activate virtualisation capability in server BIOS
3. Install KVM and LIBVIRT packages
4. Verify that KVM is working
5. Setup user and middleware application accounts
6. Mount storage array initially at ETS in Canada and secondly on gsn-pw1 server after the files were transferred
7. Configure virtual bridge for NIC that connects to GSN network
8. Install virt-manager to setup graphical interface
9. Verify KVM operations from virtual environment

The detailed version of the software installation procedure as supplied by GSN is described in *Appendix 11.2*.

5.1.1 HEAnet Nodes

HEAnet has installed five GSN nodes. Two of these had been installed before the Mantychore project, as part of the emerging collaboration with the GSN project which started slightly ahead of the Mantychore Project.

The Environmental Protection Agency (EPA) node, located in Wexford, is powered by Solar/PV panels. The server is a Dell PowerEdge R710 model; the PDU is an Avocent model, and the connection to HEAnet's network is realised through two Cisco Catalyst 2950 switches. Connectivity to this site was negotiated to use part of the client EPA's bandwidth. To minimise disruption to the client's production service, a second switch powered by a main supply was connected in parallel, so that connectivity was continuous even if access was not available to the first, solar powered switch. Figure 2 displays a solar panel array at the EPA site and Figure 3 shows the node diagram.

Figure 2 - Solar panel array at the HEAnet EPA site

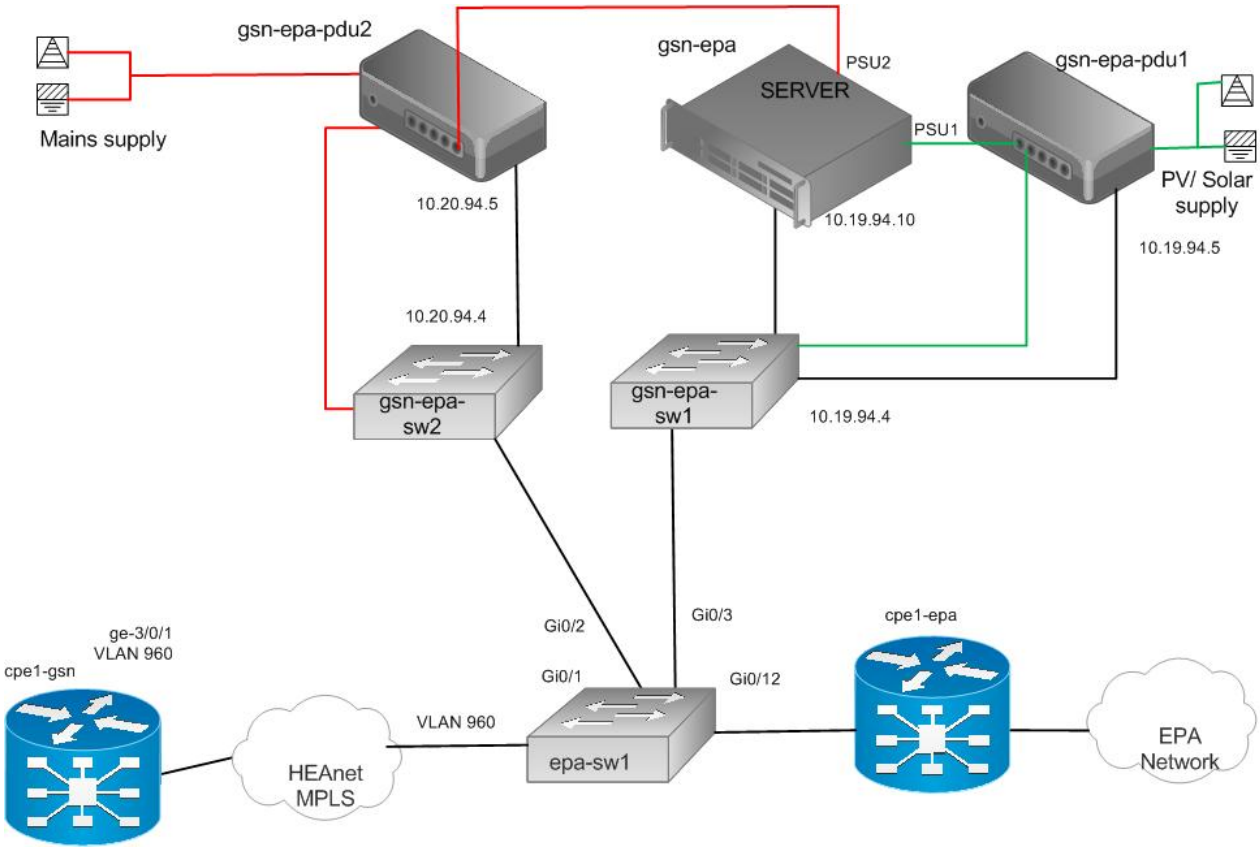

Figure 3 - HEAnet GSN node at EPA

A table was produced to indicate the average number of hours the server and switch combination was powered by the Solar/PV array. This is included in Appendix A.

The Dundalk Institute of Technology (DKIT) node – located in Dundalk – is powered by wind turbine. The server here is a Dell R710 series and the PDU is an APC model. Connectivity to the HEAnet network is achieved with a Cisco 7600 Router. The network diagram is depicted in Figure 4. Figure 5 is a picture of a wind turbine at the DKIT site.

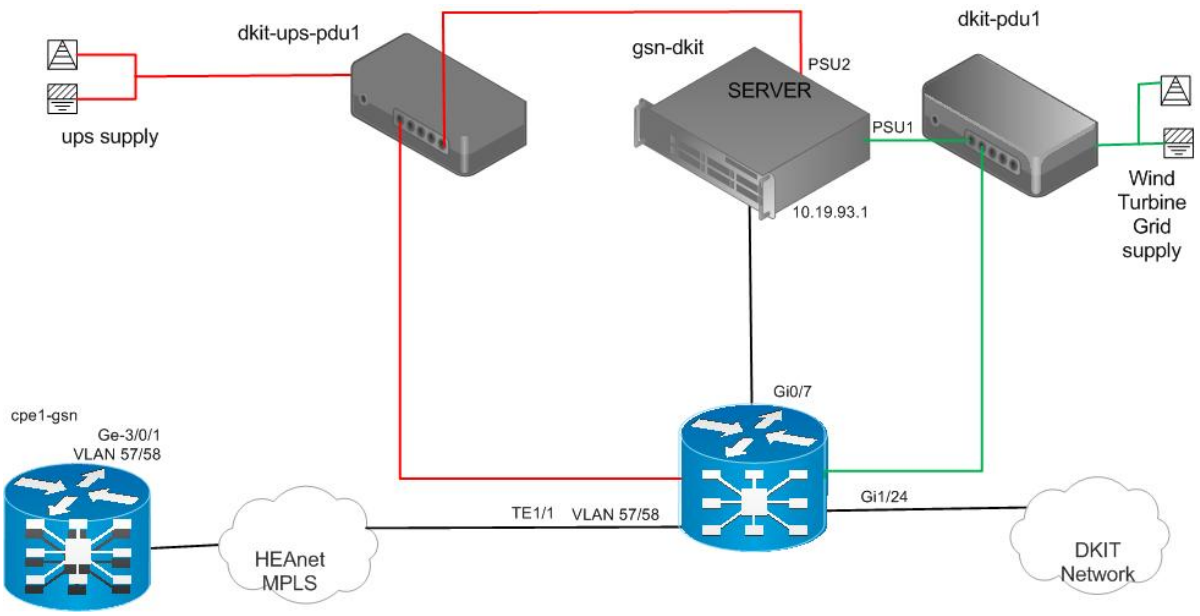

Figure 4 - HEAnet GSN node at DKIT

Figure 5 – Wind Turbine at the HEAnet DKIT site

With the beginning of the Mantychore project, HEAnet installed a third GSN node at the company headquarters in Dublin. The GD5 node is also a Dell R710 server powered by a regular AC supply but has a VARIAC attached; the server connects to both an APC and an Avocent PDU

and a Cisco Catalyst 2950 switch. This was to act as a control node and be available for experimentation and software update trials and development purposes.

For the purpose of creating a GSN hub in Europe, two more servers gsn-pw1 and gsn-pw2 were installed at the Park West site. These servers are also Dell PowerEdge R710 models, connected to cpe1-gsn router and then connected to HEAnet of concatenated switches. The PDUs here are APC models that use only a regular AC supply.

5.1.2 i2CAT Node

i2CAT in Barcelona, Spain has installed a node at the Campus Nord. Electricity comes from a Green source provider, therefore, t, there is a PDU (a Raritan DPXR8A-16 model), that is monitoring the energy consumed. Connected to the PDU there is a SMC TigerSwitch 8624T that provides connectivity to a server that follows the GSN recommendation for the Dell R710 series..

5.1.3 IBBT Node

An IBBT is not a mantychore partners, but they are an institution that joined the EU-GSN initiative at their own costs. They were initially considered for the deployment, however, the current economic situation has stop their investment on this initiative, which planned to deploy a node in Gent (Belgium) with wind power as the renewable source of energy.

5.1.4 NORDUnet Node

NORDUnet has deployed a GSN node in Reykjavík, Iceland. The node is powered, via the Icelandic grid, by hydropower (~75%) and geothermal energy (~25%). The server is a Dell PowerEdge R710 node, connected to the NORDUnet backbone through a Juniper Networks MX480 router. An Eaton PW102MA0UA64 PDU measures power consumption. The diagram in Figure 6 depicts the connectivity setup for the GSN node.

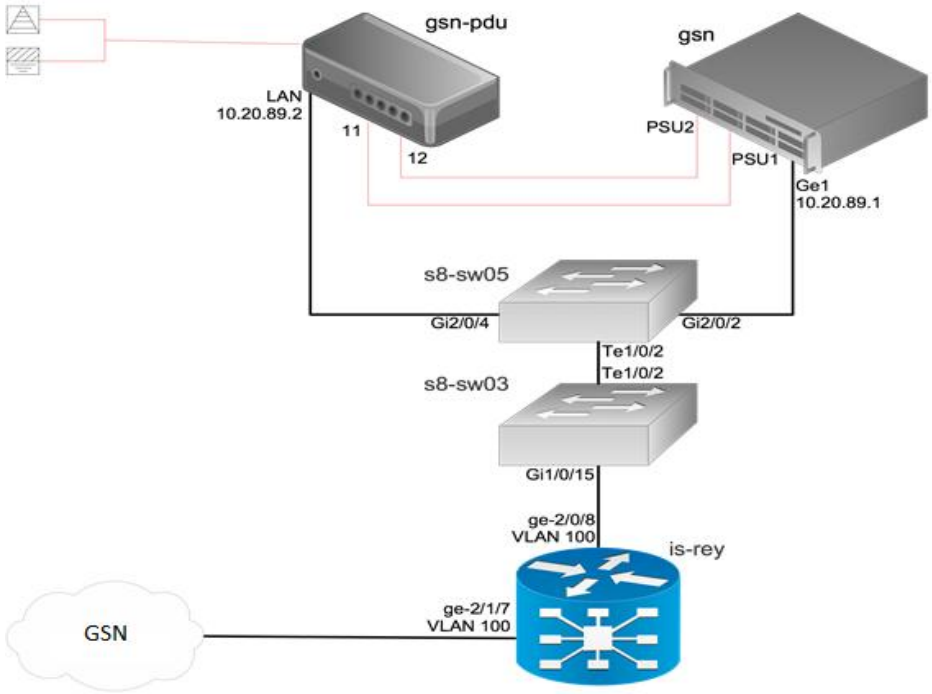

Figure 6 - NORDUnet GSN node

5.2 Mantychore/GSN Network

The integrated Mantychore/GSN network was gradually built and expanded in synchronization with the deployment of the European GSN nodes and software development effort in both projects. The original intention was to achieve a fully dynamic “Mantychore” network in Europe. However, due to external factors, the current network is based on other connectivity components as explained below. The network development consisted of three major stages, or phases.

5.2.1 Phase 1

In the first phase, the intention was to connect the GSN nodes deployed by WP7 partners directly to the Canadian GSN hub, as depicted in *Figure 7*. In reality, four European nodes were connected in this manner – three by HEAnet in Ireland and one by NORDUnet in Iceland. The HEAnet nodes were connected to Canada via GÉANT+ circuits, while the NORDUnet node was connected via a transatlantic circuit in collaboration with CANARIE.

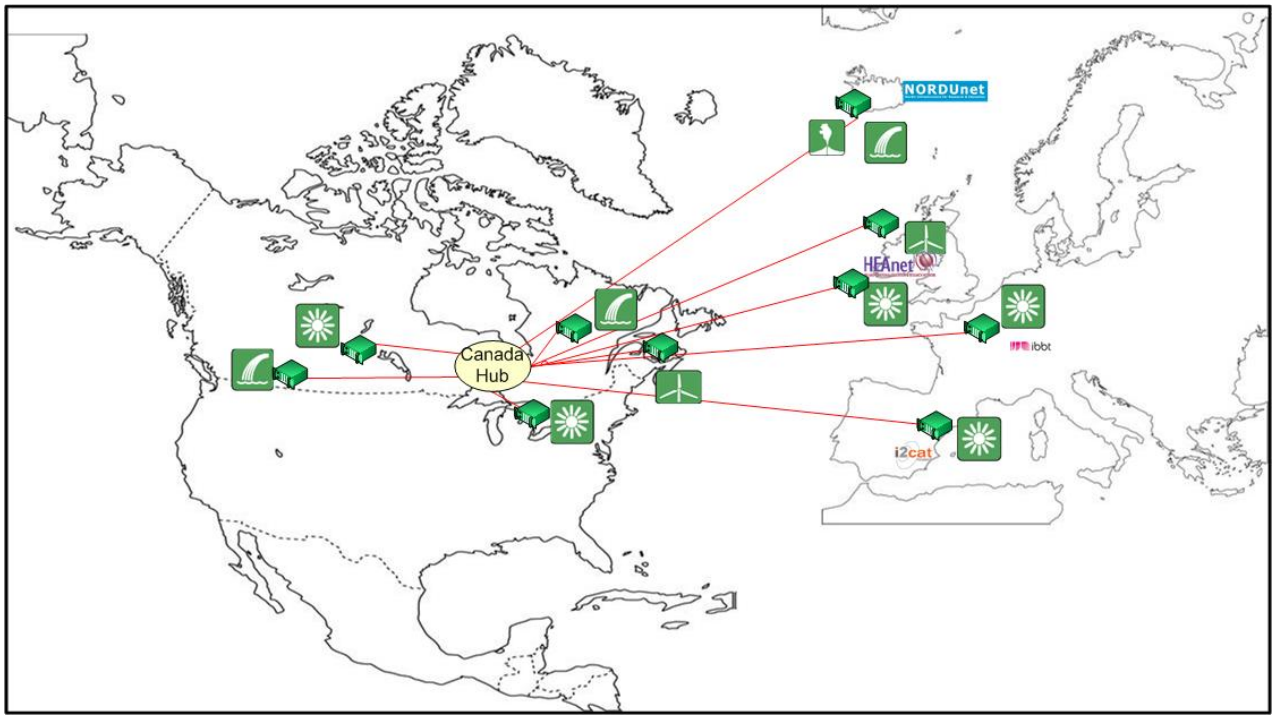

Figure 7 - Mantychore/GSN network - Phase 1[1]

5.2.2 Phase 2

In the second phase, the European GSN nodes would reorganize as an independent network (Figure 8). For this purpose, HEAnet deployed two more nodes in order to act as a European GSN hub. The rest of the HEAnet nodes were connected to the hub using HEAnet's network; the NORDUnet node was connected via a GRE tunnel; the i2CAT node was connected via a GÉANT Plus circuit.

[1] Source: www.greenstarnetwork.com

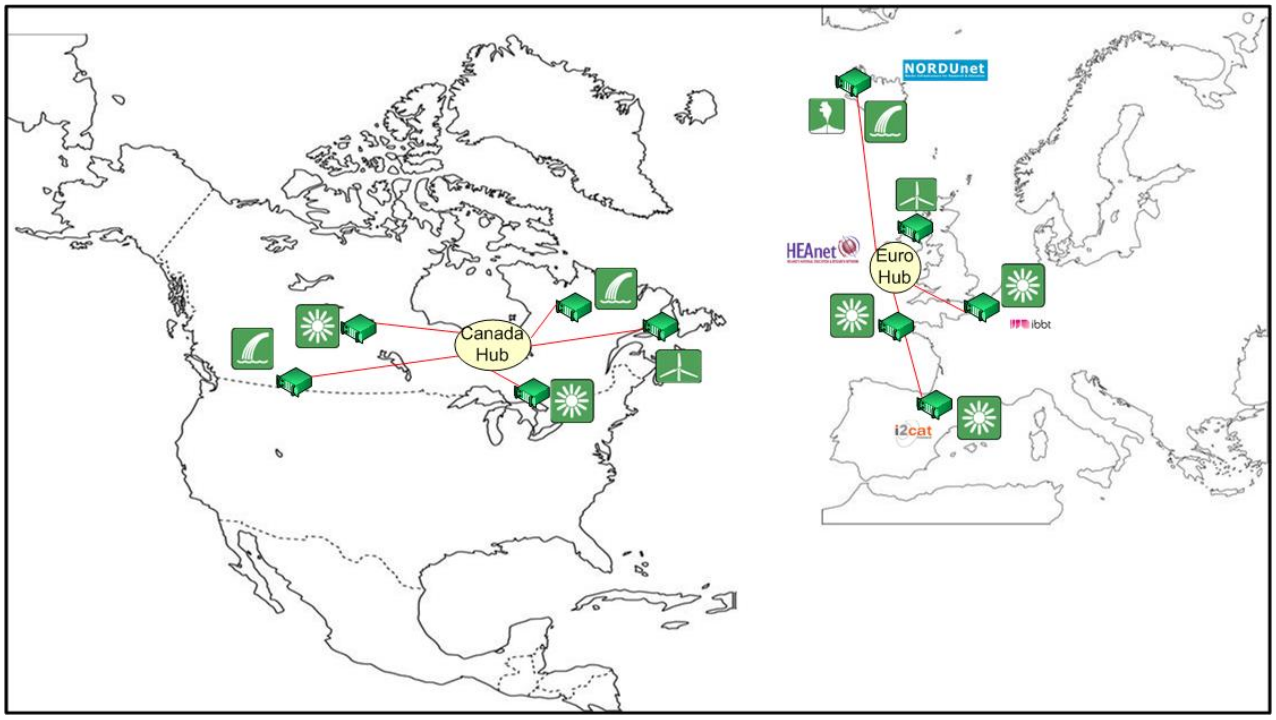

Figure 8 - Mantychore/GSN network - Phase 2[2]

5.2.3 Phase 3

The third phase would see the two networks connected together in a federated network (Figure 9). In actuality, this stage was implemented simultaneously with phase 2. The GÉANT Plus circuit previously used in phase 1 for the HEAnet nodes became the inter-network link, and the circuit previously used by the NORDUnet node became the backup connection between the two networks.

[2] Source: www.greenstarnetwork.com

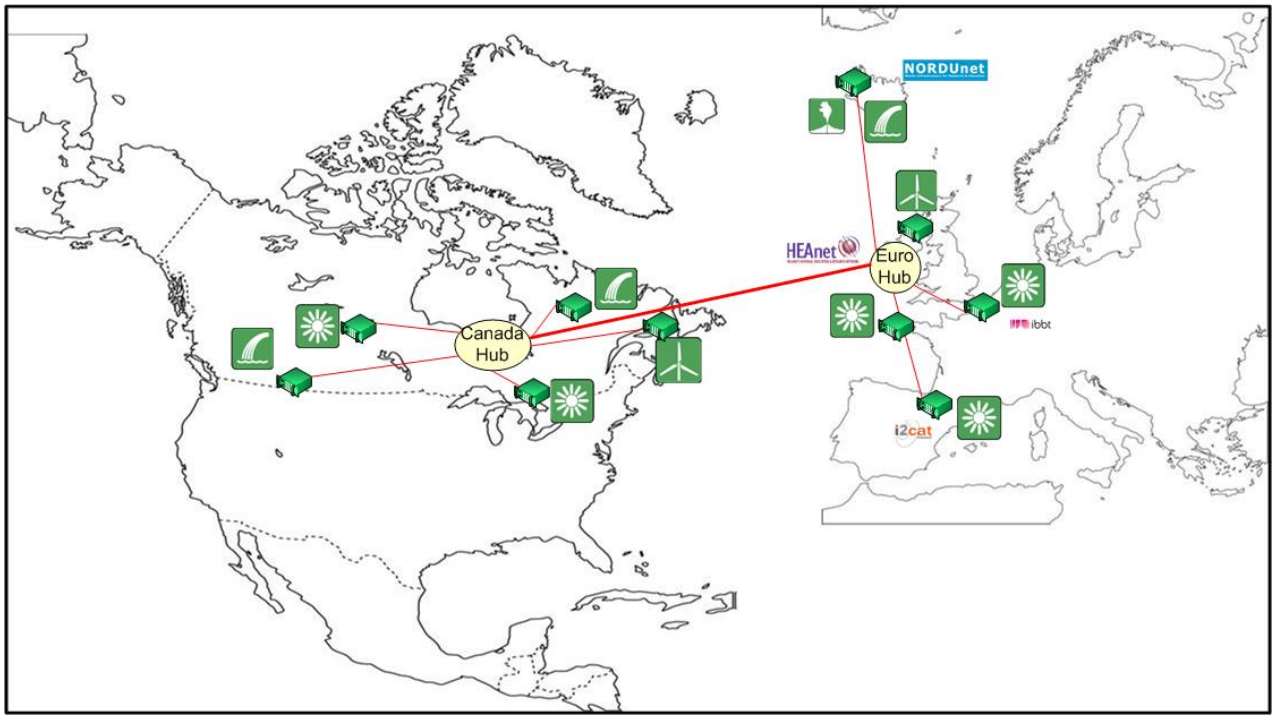

Figure 9 - Mantychore/GSN network - Phase 3[3]

5.2.4 Current Status

At the time of writing this document, the GSN/Mantychore network is provisioned as follows:

- The HEAnet nodes are connected to the local hub through HEAnet's network;
- The i2CAT node is connected to the HEAnet hub via a layer 2 GÉANT Plus circuit.
- The NORDUnet node is connected to the HEAnet hub at layer 3 over a GRE tunnel.
- The HEAnet hub connects directly to the Canadian GSN hub at both layer 2 and layer 3 over a GÉANT Plus circuit.
- There is a backup layer 2 connection between the NORDUnet node and the Canadian GSN hub. This setup will be further developed to provide an alternate hub-to-hub connection.

Figure 10 depicts the current status of the Mantychore/GSN network.

[3] Source: www.greenstarnetwork.com

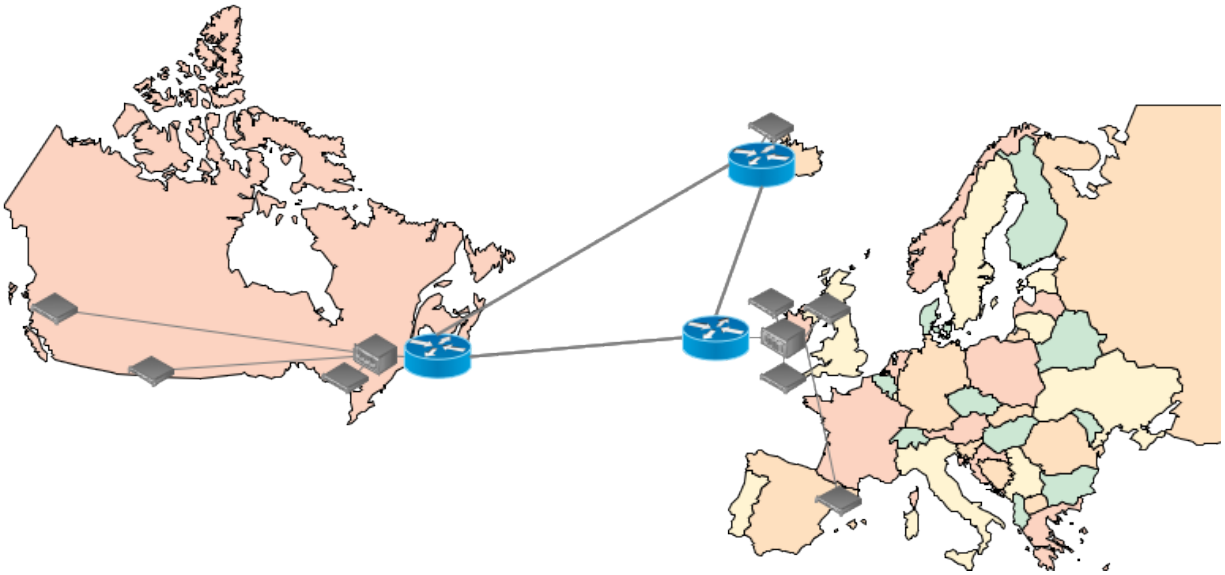

Figure 10 - Mantychore/GSN network - Current status

6 Service Integration

The service integration work in WP7 consists of three main activities. The primary activity focused on defining a use case for the interoperability of the two software frameworks, OpenNaaS and GSn respectively. The secondary activity dealt with producing a specification for the software implementation of the use case requirements. Finally, the third activity focused on extending OpenNaaS functionality with the power monitoring metrics used in GSN.

6.1 GSN Use Case

Different models and alternatives have been studied for the use case interoperability between the GSN and Mantychore software solutions. The chosen model was that of a provider–client stack, with one software solution providing a service for the other. Considering the fact that the use case was to be developed for the OpenNaaS software, it was decided that it should reflect a way in which the OpenNaaS software could provide a service for the GSN cloud services, which would then represent the client layer. As a consequence, a study was performed on GSN's networking requirements in order to detect potential points where OpenNaaS would be able to add required functionality, or improve an existing one. Achieving this use case with OpenNaaS would allow for reduced complexity and increased performance at the network layer for the GSN platform, which does not have network NaaS functionalities. Accomplishing the first scenario simplifies the one-time configuration for initial connectivity to a node, while accomplishing the second scenario helps optimise the migration process.

6.1.1 Long-term Connectivity

The first use case scenario deals with the provisioning of long-term spoke-hub connectivity, making up the "basic" GSN topology. As depicted in *Figure 11*, OpenNaaS would dynamically provision the GSN spoke-hub connectivity over (potentially) multiple heterogeneous domains, with circuits terminating in the nodes' access devices (routers or switches). The underlying network infrastructure would be transparent for the end users (i.e. the local GSN administrators). Virtual routers provisioned and controlled by Mantychore would act as gateways for the virtual data centres hosted at the GSN sites.

A number of requirements apply for this scenario:

- The end points can be in different administrative domains.
- The end points are routers or switches located at data centres hosting the GSN nodes; local access is the responsibility of the local administrators.
- The underlying infrastructure should be transparent to the users.
- IP connectivity should be tunnelled between the end points: all the nodes have to be in the same subnet and a VR should act as end point and gateway for each node and the VMs residing on it.

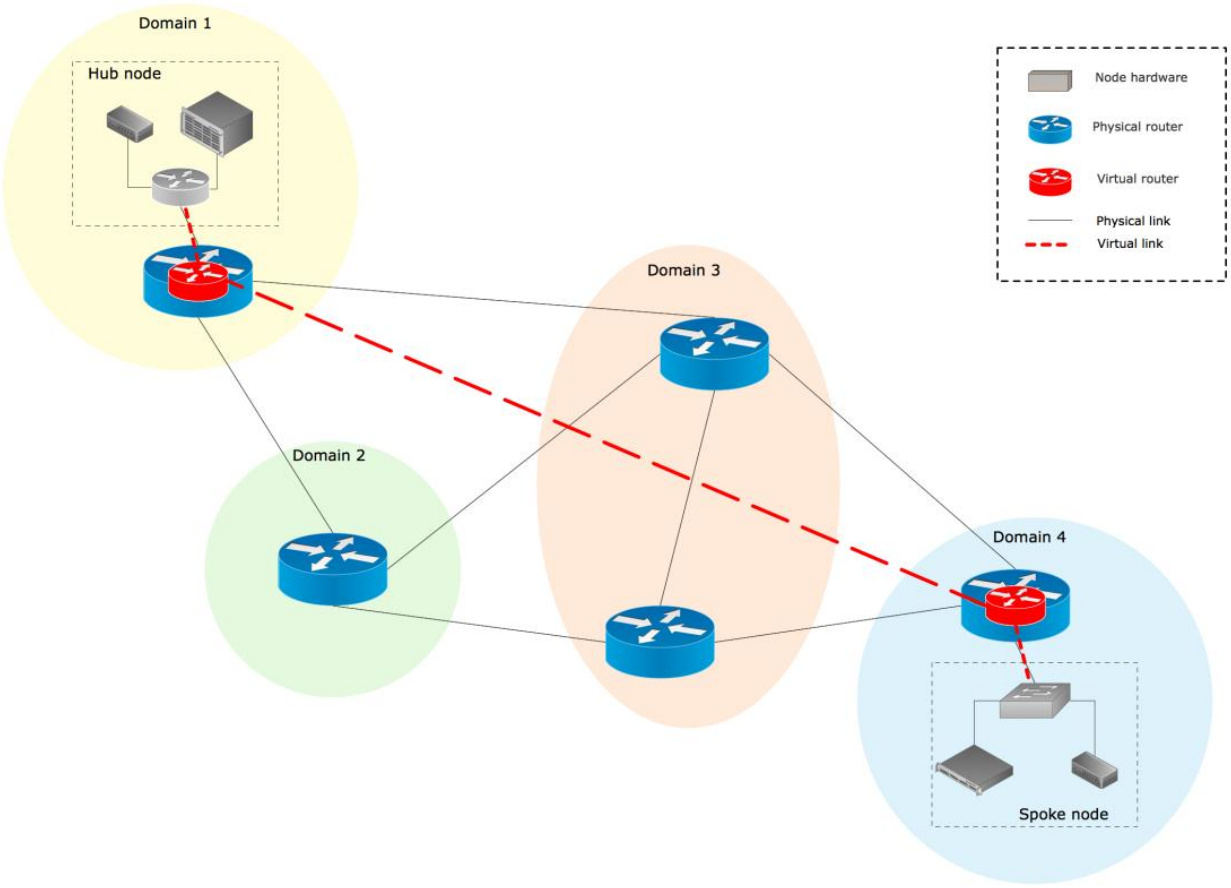

Figure 11 - GSN use case for long term connectivity

6.1.2 Temporary Connectivity

The second scenario deals with the provisioning of temporary, on-demand connectivity between GSN nodes during VM migrations. This case assumes an existing infrastructure – provisioned either statically or dynamically with OpenNaaS or another tool – connecting nodes to their local

hubs, and hubs to other hubs. When the GSN platform needs to migrate VMs from one node to another, within the same GSN cloud or in a remote cloud, it will send a call to the OpenNaaS API with the source and the destination endpoints. OpenNaaS would then provision the required elements (interfaces, links) in order to make the migration path as optimal as possible, considering also the Green parameters of GSN/ECOLO TIC. This concept is illustrated in Figure 12.

Aside from the requirements for long-term connectivity, the following constraints apply in the temporary scenario:

- Connectivity is set up between two nodes and turned down when it is no longer required.
- The VMs have to keep their IP addresses.
- Sufficient bandwidth has to be available between the end points for the entire duration of the migration.

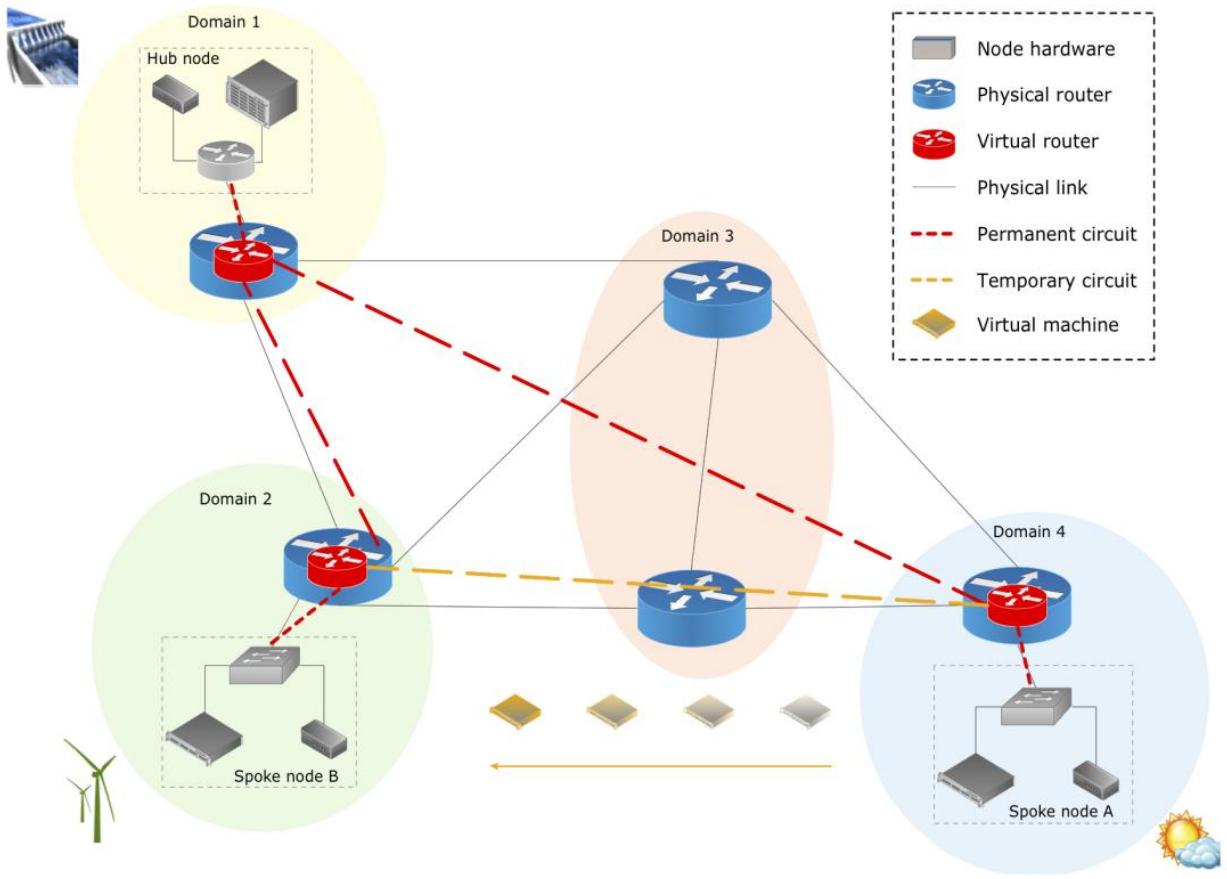

Figure 12 - GSN use case for temporary connectivity

6.2 API Design

The work on the API consisted of describing and defining the interaction between the two software platforms. This was done by a simplified test case focusing on the details of the inter-software communication. For this purpose, a scenario has been produced for the case of inter-cloud non-live migration. The steps of this process, along with the related software components are described below.

1. The *GSN Cloud Manager* decides that a certain VM needs to be migrated, based on manual configuration, or on information received from the *GSN Facility Manager;*
2. The *GSN Cloud Manager* chooses the destination of the VM to be a foreign GSN cloud;
3. The *GSN Cloud Manager* sends the source and destination identifiers (IP addresses) to the *GSN Network Manager;*
4. The *GSN Network Manager* generates a circuit request to the *OpenNaaS API,* containing the two endpoints along with other optional parameters, such as bandwidth;
5. *OpenNaaS* configures the required circuit (tunnel) by means of configuring logical routers and interfaces on the routers that it controls, and then returns a circuit ID to the *GSN Network Manager;*
6. The *GSN Network Manager* informs the *GSN Cloud Manager* of the availability of the required connectivity;
7. The *GSN Cloud Manager* performs the VM migration, and then informs the *GSN Network Manager* that the operation is complete;
8. The *GSN Network Manager* sends a termination request to the *OpenNaaS API;*
9. *OpenNaaS* deprovisions the previously configured tunnel.

Figure 13 illustrates an overview of the messaging between the software components involved. The diagram also highlights the main functions of the software components that are centric to the platform integration.

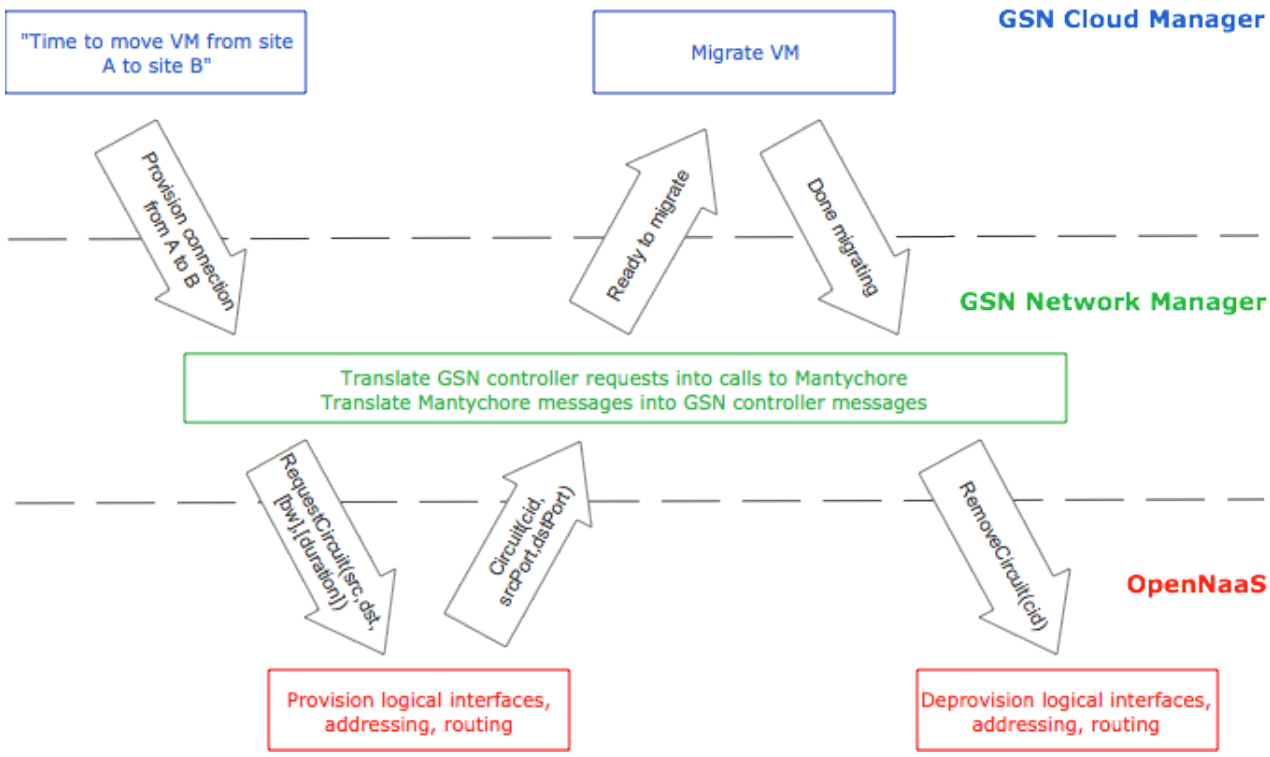
"Time to move VM from site A to site B"
Migrate VM
GSN Cloud Manager
Provision connection from A to B
Ready to migrate
Done migrating
GSN Network Manager
Translate GSN controller requests into calls to Mantychore
Translate Mantychore messages into GSN controller messages
RequestCircuit(src,dst, [bw],[duration])
CircuitId(cid, srcPort,dstPort)
RemoveCircuit(cid)
OpenNaaS
Provision logical interfaces, addressing, routing
Deprovision logical interfaces, addressing, routing

Figure 13 - OpenNaaS - GSN messaging sequence

6.3 Integration of Power Monitoring Metrics

6.3.1 Overview

During OpenNaaS and GSN Cloud Manager Interoperability, some components of the latter have been identified as very valuable for OpenNaaS. These components are the ones related to gathering power consumption statistics from capable devices; specifically the PDU resource together with the Raritan PCR8 driver. Some of the work of WP7 services integration has been focused on porting this driver to OpenNaaS.

As mentioned, the GSN software is able to control a Raritan PCR8 device [6], which is capable of:

- Powering on and off plugged resources,
- Measuring power consumption statistics per plugged resource.

GSN software internally uses this resource but does not expose its capabilities directly to the user. That is what OpenNaaS will offer. OpenNaaS plans to integrate the GSN Raritan PCR8 driver and export its capabilities to other resources. Power management (power on/power off) and power monitoring capabilities will be exposed to the user, although the existence of Raritan PCR8 devices may not be visible. Additionally, the power-monitoring capability will be enhanced to maintain a history of read values, in order to provide an overview of consumption over time.

With these capabilities in place, OpenNaaS clients (both users and other applications) would be able to consider power consumption information in their network allocation procedures and even automatically power on and off resources based on the need to reduce environmental costs.

6.3.2 OpenNaaS Power Monitoring and Management API

OpenNaaS plans to export two capabilities:

1. Power switching: focused on powering resources on and off,
2. Power monitoring: focused on retrieving power consumption information from resources.

The power switching capability will have two main operations: power on and power off. Other operations derived from these ones may exist (e.g. toggle power or reboot). However, at this point it cannot be assured they will be part of the API.

The power monitoring capability will take power consumption measurements at a user specified constant rate and maintain a history with them. Available power consumption measurements will include current (Amperes), voltage (Volts) and power (Watts). Exported operations will allow reading the history (getHistory), clear the history (clearHistory) and get last measurement (getCurrentMeasurement). Other operations may exist such as filtering getHistory output (e.g. getHistoryOlderThan(date), getHistoryLastMeasurement(int number)). However at this point it cannot be assured they will be part of the API.

These capabilities, or the drivers, may require per instance configuration, which can be achieved by specifying key-value pairs in the capability descriptor (e.g. refresh rate). Available keys and valid domains for values will be published in driver documentation.

As already mentioned, these capabilities will require the existence of a driver supporting them. Here is where the Raritan PCR8 driver from GSN plays its part.

6.3.3 Portability details

In order to support the capabilities defined in previous section, the Raritan PCR8 driver from GSN must be integrated into OpenNaaS.

To manage Raritan PCR8 units, the GSN software defines a Raritan PCR8 resource with a generic PDU model. This resource is linked to an action set that uses a CLI protocol over a Telnet session to communicate with a PCR8 unit. This action set consists on the following actions:

These actions match well with capabilities defined by OpenNaaS. This way, On, Off and Toggle actions would be part of the power switching capability, while QueryOutlet, ShowOutlet and Show Sensors actions would be part of the power monitoring capability. The rest of the actions may not be part of described capabilities, although they have their use when managing a PDU resource, and may be ported as well.

Capabilities and actions will need a data model to retrieve and store valuable information. The fact that the GSN PDU model is generic makes it reusable for other PDU drivers, and therefore a good candidate to be used as the data model for the capabilities of any PDU resource.

However, the fact that GSN does not use generic resources, but fabric dependent ones, makes the Raritan PCR8 resource not directly importable into OpenNaaS. It has to be decided if OpenNaaS will show the PDU as an independent resource with its own capabilities and model, or if exposed capabilities will be part of other resources and the fact of using a PDU will be left to the driver implementation. In the second case, a question yet to be answered is how to integrate PDU measurements into these other resources data model.

Last, but not least, protocols used by Raritan PCR8 driver should be also ported to OpenNaaS and integrated with its ProtocolSessionManager.

6.3.4 Future extensions

This extension leaves the door open to the integration of other PDU models in OpenNaaS. The fact of using a generic model and the architecture of OpenNaaS itself is meant to ease integration with other resources, especially if resources of the same type already exist in OpenNaaS. In this sense, it is expected to be relatively easy to build new drivers for APC or Avocent PDUs, among others, especially if they make use of protocols already existing in OpenNaaS.

7 Liaisons

The work in T7.1 has been performed in close collaboration with the GSN project and, as a consequence, a number of organisations were involved that are not part of the Mantychore project. CANARIE – the Canadian NREN – made a considerable contribution to the physical connectivity to the GSN nodes in Canada. Communications Research Centre Canada (CRC) was engaged in matters pertaining to software integration, as well as data acquisition. Finally, the Synchromedia department of École de Technologie Supérieure (ETS) was involved in matters of publishing and dissemination as well as overall project guidance. This liaison between the Mantychore and GSN projects was acknowledged on the websites of both projects [7, 8] and led to the publication of [5]. Actually, and as already commented previously, due to the fact the GSN project has finished, the relationship continues with the partners belonging to GSN which are participating in the new phase of the GSN project, called ECOLO TIC. For that purpose, a MoU has been signed.

Toward the end of the activity a liaison was initiated with the GÉANT Bandwidth-on-Demand (BoD) and environmental activities. The activity in WP7 generated interest from the green community in GÉANT and this led to the integration of BoD services in the provisioning of connectivity between GSN nodes in Europe [9].

8 Conclusion

This report has presented the steps taken towards the interoperability of Mantychore and GSN services. Two major areas of copperation were identified and addressed: network integration and service compatibility.

HEAnet, i2CAT and NORDUnet have deployed a number of GSN nodes in their networks following the specifications received from the GSN project team. These nodes are powered by different sources of renewable energy, such as wind power, solar power or geothermal energy. The GSN software has been installed on these nodes in order to firstly integrate them into the Canadian GSN cloud, and then to reorganise them in a self-contained European cloud. These nodes were first connected physically to the Canadian network, and later to each other, while still maintaining connectivity to GSN in Canada. Presently there are seven operational European GSN nodes: five in HEAnet and one each in i2CAT and NORDUnet. The next steps are to establish routines for the use of these nodes and scheduled migrations between them in order to obtain power data. Later on, OpenNaaS has become the controller of these nodes.

Part of the work in T7.1 was to identify potential use cases for software integration between the Mantychore and GSN solutions. The node deployment work described above helped reveal two such use cases, where OpenNaaS would provide a network service to GSN in order to streamline and optimise network provisioning. In the first use case, GSN would request OpenNaaS to provision long term layer 2 connections between spoke nodes and hub nodes in the GSN cloud. The second use case describes the provisioning and de-provisioning of short-term connections with OpenNaaS for virtual machine (VM) migrations between any two (or more) GSN nodes. As a follow-up to the definition of these use cases, a high-level design was produced for an API that would be developed for OpenNaaS. During the software study phase a number of GSN features have been identified as potentially valuable for OpenNaaS. One such example is the power measurement and management function. This will be integrated in OpenNaaS as part of a development task. A generic service request API that would address both previously mentioned use cases is also considered for implementation.

Being a collaboration task, the work involved external parties, including as the GSN consortium members, as well as GÉANT. This led to joint publications and a considerable amount of knowledge transfer.

The work on WP7 will continue, with its focus shifted from the deployment work to measurement and optimisation activities. The service integration will continue with the development and refining of OpenNaaS' energy features according to GSN and ECOLO TIC requirements

9 References

[1] MANTYCHORE Seventh Framework Programme, Annex 1 – "Description of Work". (2011).

[2] D. O'Callaghan, S. Figuerola, P. Minoves. (2011). *Mantychore – Dynamic networks for e-Infrastructure.* EGI User Forum 2011.

[3] M. Brooks, M. Lemay, M. Savoie, R. Simmonds, C. Kiddle, S. Campbell. (2010). *All About GSN.*

[4] W. Van Heddeghem, W. Vereecken, D. Colle, M. Pickavet, P. Demeester. (2011). *Distributed Computing for Footprint Reduction by Exploiting Low-Footprint Energy Availability.* Future Generation Computer Systems, Green Computing special issue, May 2011

[5] K.K. Nguyen, M. Cheriet, M. Lemay, B. St. Arnaud, V. Reijs, A. Mackarel, P. Minoves, A. Pastrama, W. Van Heddeghem. (2011). *Renewable Energy Provisioning for ICT Services in a Future Internet.* Lecture Notes in Computer Science, 2011, Volume 6656/2011, p. 419-429.

[6] Raritan RPC. http://www.raritan.com/support/end-of-life/remote-power-control/ (Visited: 16/08/2012).

[7] Mantychore Project – *Collaborations*, http://www.mantychore.eu/collaborations/ (Visited: 30/08/2012).

[8] Greenstar Network – *Collaboration Mantychore FP7*, http://greenstarnetwork.com/index.php/en/collaboration/mantychore-fp7 (Visited: 30/08/2012).

[9] GÉANT – *GreenStar project and GÉANT demonstrate potential for green networking*, http://www.geant.net/Media_Centre/News/Pages/GreenStar-project-and-GEANT-demonstrate-potential-for-green-networking.aspx (19/07/2012).

10 Acronyms

API	Application Programming Interface
CLI	Command Line Interface
DKIT	Dundalk Institute of Technology
EPA	Environmental Protection Agency
GRE	Generic Routing Encapsulation
GSN	GreenStar Network
IaaS	Infrastructure as a Service
PDU	Power Distribution Unit
VARIAC	Variable Autotransformer
VM	Virtual Machine
VR	Virtual Router

11 Appendices

11.1 Solar Power Generation at EPA

Month	Average Hours Generation Per Day	Hours Range across Month	
		Minimum	Maximum
Nov	4	2,75	7
Dec	5	0	6,25
Jan	5	0	7
Feb	6	0	8,25
March	9	6,75	10,75
April	11	9	12,5
May	11	9,75	13
June	13	10	14,25
July	13	9,25	14,25
Aug	12	10,25	13,25
Sep	10	8,25	11,5
Oct	8	3,25	9,25

11.2 GSN Host Installation Process

1. Install the operating system. Ubuntu 10.04 LTS 64-bit OS.

Update Ubuntu's list of available packages:

apt-get update

Upgrade the currently installed packages to the most recent versions:

apt-get upgrade

2. Activate the virtualization capability of PCs. This function is found in the "Performance" category of the setup menu of PCs (press F2 or F12 when booting a PC).

[It may be in Virtualization Support -> Enable Intel Virtualization Technology.]

3. Install kvm:

sudo apt-get install kvm

4. Install libvirt:

sudo apt-get install libvirt-bin

5. Verify that KVM is ok:

kvm-ok

If you get the following message you are good to go, otherwise something is not configured or the BIOS is not set for HW Virtualization:

INFO: Your CPU supports KVM extensions

INFO: /dev/kvm exists

KVM acceleration can be used

6. Create an admin account (with sudo permissions) for management:

sudo adduser –system system

sudo passwd system

7. Create an account for middleware application:

sudo adduser –force GsnM-ws-VmmEngine

Add this user to fuse, libvirtd, and admin groups:

sudo adduser GsnM-ws-VmmEngine libvertd

sudo adduser GsnM-ws-VmmEngine fuse

sudo adduser GsnM-ws-VmmEngine admin

8. Mounting the Storage Array Located at ÉTS:

Install sshfs package:

sudo apt-get install sshfs

Create mountpoint with all permissions for GsnM-ws-VmmEngine account:

sudo mkdir /storage

sudo chown GsnM-ws-VmmEngine /storage

Mount the storage /space/r0 for GsnM-ws-VmmEngine account

sudo -u GsnM-ws-VmmEngine sshfs -o idmap=user GsnM-ws-VmmEngine@10.20.100.7:/space/r0 /storage -o allow_root -o ServerAliveInterval=10 -o ServerAliveCountMax=3 –o reconnect

Put this command in /etc/fstab and /etc/rc.local so the storage will be mounted automatically when rebooting the host.

Add storage hostname (“10.20.100.7 storage1.gsn.local storage1”) to /etc/hosts because libvirt does not recognize IP addresses.

Configure password-less SSh to the storage server (10.20.100.7)

ssh-keygen -trsa -N '' (two single quotations at the end)

ssh-copy-id-i /home/GsnM-ws-VmmEngine/.ssh/id_rsa.pub GsnM-ws-VmmEngine@10.20.100.7

SSH the storage server

ssh GsnM-ws-VmmEngine@10.20.100.7

It will ask for the password one time only. This will add the key to the authorized keys file.

9. Configure a virtual bridge (br0) for NIC which is connected to GSN network

10. To allow migration with other hosts make sure:

a) Both hosts appear in the /etc/hosts of each other, because by default libvirt does not recognize IP addresses.

b) Set a password-less SSH for GsnM-ws-VmmEngine account in 2 directions.

No need to regenerate the key (use the one which generated in step 8):

ssh-copy-id-i /home/GsnM-ws-VmmEngine/.ssh/id_rsa.pub GsnM-ws-VmmEngine@hostIP

Connect to this host and copy the public key for GsnM-ws-VmmEngine to the new host:

ssh-copy-id-i /home/GsnM-ws-VmmEngine/.ssh/id_rsa.pub GsnM-ws-VmmEngine@IP_of_the_new_host

Verify the password-less SSH is working in both sides.

11. Install virt-manager in order to have a graphical interface.

The current version of virt-manager has some bugs when moving VMs. So it would be better to install virt-manager 0.8.4: http://pyl.pylanglois.com/2010/05/12/how-i-installed-virt-manager-0-8-4-on-ubuntu-10-04/

12. Verify KVM operations using virsh commands.

Login with GsnM-ws-VmmEngine

cd /storage

virsh

virsh# create vmname.xml

virsh# list

Id Name State

---

1 vmname running

virsh#migrate vmname qemu+ssh://remotehost/system